club commodore

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

Boletín informativo para los usuarios de microordenadores

VIC y CBM

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□



PRINTER PLOTTER 1520



enero 1984

DESCRIPCIÓN ALFABÉTICA DE LOS MNEMÓNICOS DEL 6502/6510 (I)

ADC — ADC

Suma la memoria al acumulador con acarreo

Operación: A + M + C → A, C

(Ref.: 2.2.1)

N Z C I D V
√ √ √ – – √

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Inmediato	ADC →Oper.	69	2	2
Pág. Cero	ADC Oper.	65	2	3
Pág. Cero, X	ADC Oper., X	75	2	4
Absoluto	ADC Oper.	6D	3	4
Absoluto, X	ADC Oper., X	7D	3	4*
Absoluto, Y	ADC Oper., Y	79	3	4*
(Indir., X)	ADC (Oper., X)	61	2	6
(Indir.), Y	ADC (Oper.), Y	71	2	5*

* Suma 1 si se cambia de página.

AND — AND

AND lógico con el acumulador

Operación: A ∧ M → A

(Ref.: 2.2.3.0)

N Z C I D V
√ √ – – – –

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Inmediato	AND →Oper.	29	2	2
Pág. Cero	AND Oper.	25	2	3
Pág. Cero, X	AND Oper., X	35	2	4
Absoluto	AND Oper.	2D	3	4
Absoluto, X	AND Oper., X	3D	3	4*
Absoluto, Y	AND Oper., Y	39	3	4*
(Indir., X)	AND (Oper, X)	21	2	6
(Indir.), Y	AND (Indir), Y	31	2	5

* Suma 1 si se cambia de página.

ASL — ASL

Cambia el bit izquierdo

Operación: C ← | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Ø | ← Ø

(Ref.: 10.2)

N Z C I D V
√ √ √ – – –

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Acumulador	ASL A	0A	1	2
Pág. Cero	ASL Oper.	06	2	5
Pág. Cero, X	ASL Oper., X	16	2	6
Absoluto	ASL Oper.	0E	3	6
Absoluto, X	ASL Oper., X	1E	3	7

BCC — BCC

Salta si acarreo desactivado

Operación: Salta si C = 0

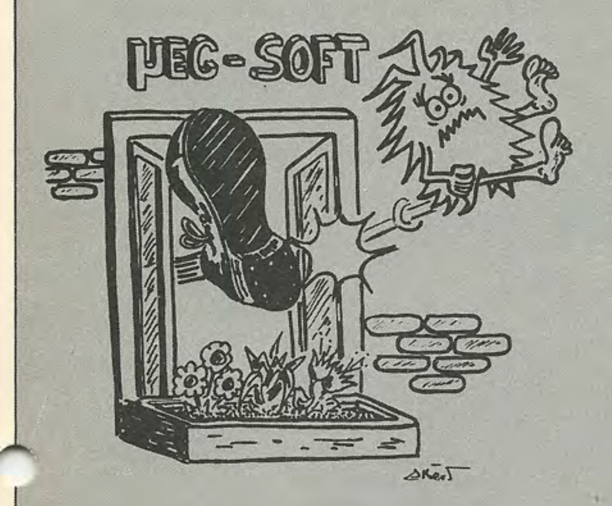
(Ref.: 4.1.1.3)

N Z C I D V
– – – – – –

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
RELATIVO	BCC Oper.	90	2	2*

* Suma 1 si el salto va a la misma página.
* Suma 2 si se salta a otra página.

Depósito Legal Sep. B. 2133-1958 - Altés, s.l., Caballero 87, Barcelona



VENTANA CBM

READY (del chip a la base de datos) (3)

por **RAFAEL NAVARRO** (M.E.C. SOFT.)

Tal y como se comentó con anterioridad, desde el punto de vista del software, el sistema operativo es el alma del ordenador. Es él quien se encarga de comunicarle con el exterior, es decir, con la pantalla, la memoria y el teclado —periféricos fundamentales— y los otros periféricos (impresoras, plotters, disquetes, discos, cassettes, etc., etc.).

El conjunto de rutinas que componen los sistemas operativos integran un cuerpo básico de operaciones elementales constituyendo, a nivel de sistema, lo mismo que las microinstrucciones a nivel de microprocesador (ver parte I de esta serie). Así, el software de base (lenguaje BASIC, por ejemplo), utiliza rutinas del sistema operativo en la ejecución de sus comandos (PRINT, por ejemplo, como se verá más adelante) del mismo modo que el código máquina necesita de las microinstrucciones del microprocesador para ser ejecutado. Queda claro pues que aunque los lenguajes y utilitarios utilicen el sistema operativo, existe una notable diferencia entre dichas entidades y sus niveles de actuación: el sistema operativo es el elemento fundamental sin el cual los otros no son **operativos.**

Más adelante, analizaremos algunos sistemas operativos para microordenadores; unos propios (CP/M, MS/DOS, etc.) y otros adaptados de los existentes para minis (UNIX, XENIX, OASIS, BDOS, etc.). Ahora nos detendremos —sin partidis-

(continúa en la pág. siguiente)

TABLA 1.1. RUTINAS KERNAL DE MANEJO DE MEMORIA

Nombre	Dirección Hexadecimal	Decimal	Uso
IOBASE	$FFF3	65523	Devuelve la dirección de base de los periféricos I/O.
MEMBOT	$FF9C	65436	Lee/fija la dirección del menor byte de RAM.
MEMTOP	$FF99	65433	Lee/fija la dirección del byte más alto de RAM.

TABLA 1.2. RUTINAS KERNAL DEL TECLADO

Nombre	Dirección Hexadecimal	Decimal	Uso
GETIN	$FFE4 (I)	65512	Coge un carácter desde el «buffer» del teclado.
SCNKEY	$FF9F	65439	Explora el teclado y pone los valores en el «buffer» del teclado.
STOP	$FFE1 (I)	65505	Chequea la tecla STOP.

EDITORIAL

sólo cuatro líneas

Este mes el editorial es especialmente corto. Sólo queremos expresar nuestro contento porque hemos recibido suficientes colaboraciones para llenar un número de la Revista. ¡Gracias a los colaboradores presentes y futuros y a ver si cunde el ejemplo!

VENTANA CBM

READY (del chip a la base de datos)

(continuación)

mos — en el desarrollado por COMMODORE.

EL KERNAL

Uno de los principales problemas de cara a los programadores en el campo de los micro-ordenadores es el miedo a las modificaciones que el fabricante pueda hacer en el ordenador. Los programas en lenguaje máquina no durarían mucho tiempo, obligando a continuas revisiones. Commodore ha desarrollado un conjunto de rutinas llamadas **rutinas KERNAL** (sistema operativo) para evitar que el programador tenga que realizar gran cantidad de cambios cuando se provee de una nueva versión del sistema operativo.

A dichas rutinas de sistema operativo se accede a través de una tabla de saltos standard. Estas rutinas se encargan del procesamiento de las funciones de entrada/salida y de manejo de memoria.

La tabla de saltos Kernal está localizada en una sección protegida de la memoria del ordenador. Cuando se añaden nuevas rutinas Kernal, la tabla de saltos es actualizada automáticamente.

Si se usan las rutinas Kernal al escribir un programa en código máquina, no hay que preocuparse por el hecho de que futuras modificaciones del sistema operativo hayan de provocar la reescritura de la mayor parte del programa. Cualquier cambio se reflejará en las nuevas rutinas Kernal que acompañan a la nueva versión del sistema operativo no provocando la reescritura de los programas, sino simplemente su reensamblaje.

1.1. CLASIFICACIÓN DE LAS RUTINAS KERNAL

Todas las rutinas Kernal están estructuradas como **subrutinas** que se

TABLA 1.3. RUTINAS KERNAL DE PANTALLA

Nombre	Dirección Hexadecimal	Decimal	Uso
PLOT	$FFF0	65520	Lee/fija la corriente posición X,Y del cursor en pantalla.
SCREEN	$FFED	65517	Devuelve la organización de la pantalla como el número de columnas (X) y filas (Y).

TABLA 1.4. RUTINAS KERNAL DEL RELOJ

Nombre	Dirección Hexadecimal	Decimal	Uso
RDTIM	$FFDE	65502	Lee el reloj del sistema que devuelve AM/PM, horas, minutos, segundos, y décimas de segundo.
SETTIM	$FFDB	65499	Fija el reloj del sistema.
UDTIM	$FFEA	65514	Incrementa el reloj del sistema (normalmente usada por las rutinas del sistema de interrupción).

TABLA 1.5. RUTINAS KERNAL DE CONTROL DE PERIFÉRICOS

Nombre	Dirección Hexadecimal	Decimal	Uso
LKUPLA	$FF8D	65415	Definir los parámetros de fichero basados en la información de dirección lógica.
LKUPSA	$FF8A	65418	Definir los parámetros de fichero basados en la información de la dirección secundaria.
SETLFS	$FFBA	65466	Fijar el número de fichero lógico, número de periférico, e información del comando.
SETNAM	$FFBD	65469	Fijar la información del nombre de fichero.

TABLA 1.6. RUTINAS KERNAL DEL BUS IEEE

Nombre	Dirección Hexadecimal	Decimal	Uso
ACPTR	$FFA5	65445	Acepta un byte de entrada desde el IEEE con total «handshaking».
CIOUT	$FFA8	65448	Envía un byte al IEEE con total «handshaking».
LISTEN	$FFB1	65457	Escucha con atención en el IEE bus.
SECOND	$FF93	65427	Envía un comando secundario después de LISTEN.
SETTMO	$FFA2	65442	Activar/desactivar fueras de tiempo (timeouts) sobre el bus IEEE.
TALK	$FFB4	65460	Enviar una dirección de escucha al IEEE.
TKSA	$FF96	65430	Enviar una dirección secundaria para escucha (TALK).
UNLSN	$FFAE	65454	Comanda el IEEE a UNLISTEN (cierra el IEEE para escucha).
UNTLK	$FFAB	65451	Comanda el IEEE a UNTALK (cierra para diálogo).

VENTANA CBM

llaman desde su programa, suministrándole información a algunas rutinas y recibiendo información desde otras. Ello significa que se llamará la rutina con una instrucción JSR. La rutina finaliza con una instrucción RTS y vuelve a la instrucción siguiente a la instrucción JSR que la llamó. «Microelectrónica y Control» tiene editados varios manuales en los que se detalla la utilización del KERNAL:

— rutinas de tratamiento de memoria;
— rutinas de teclado;
— rutinas de pantalla;
— rutinas del sistema de reloj;
— rutinas de control de periféricos;
— rutinas de control de ficheros;
— rutinas del bus IEEE;
— rutinas de canal;
— rutinas de vector;
— rutinas de control del sistema;
— rutinas de dirección de memoria /periférico.

Estas categorías se establecen simplemente para tener una visión de los tipos de funciones disponibles a través de las rutinas Kernal. Algunas rutinas, especialmente las que interaccionan con los periféricos y el bus IEE-488 se extienden en varias categorías. Las tablas 1.1 hasta 1.11 muestran las rutinas de cada categoría. Aquellas que son llamadas indirectamente están marcadas con un «(I)» después de la dirección hexadecimal.

TABLA 1.11. CÓDIGOS DE ERROR DEL KERNAL

Código error	significado
0	Rutina interrumpida por la pulsación de STOP
1	Demasiados ficheros abiertos
2	Fichero ya abierto
3	Fichero no abierto
4	Fichero no encontrado
5	Periférico no presente
6	No es fichero de entrada (y se intenta input)
7	No es fichero de salida (y se intenta output)
8	Se ha omitido el nombre del fichero
9	Número de perif. ilegal

TABLA 1.7. RUTINAS KERNAL DE CANAL

Nombre	Dirección Hexadecimal	Decimal	Uso
CHKIN	$FFC6 (I)	65478	Abre un canal para entrada.
CHKOUT	$FFC9 (I)	65481	Abre un canal para salida.
CHRIN	$FFCF (I)	65487	Acepta un carácter de entrada desde el canal.
CHROUT	$FFD2 (I)	65490	Envía un carácter de salida al canal.
CLRCHN	$FFCC (I)	65484	Borra todos los canales. Cierra todos los canales abiertos y restablece los canales por defecto.

TABLA 1.8. RUTINAS KERNAL DE VECTOR

Nombre	Dirección Hexadecimal	Decimal	Uso
RESTOR	$FF87	65415	Restablece los valores por defecto de los vectores de las subrutinas del sistema e interrupciones.
VECTOR	$FF84	65412	Lee/fija el contenido actual de los vectores RAM del sistema.

TABLA 1.9. RUTINAS KERNAL DE CONTROL DEL SISTEMA

Nombre	Dirección Hexadecimal	Decimal	Uso
READST	$FFB7	65463	Lee la palabra de estado del sistema.
SETMSG	$FF90	65424	Control de impresión de los mensajes de error Kernal y los mensajes de diagnóstico.

TABLA 1.10. RUTINAS KERNAL DIRECTAS MEMORIA/PERIFÉRICO

Nombre	Dirección Hexadecimal	Decimal	Uso
LOAD	$FFD5 (I)	65493	Carga directa desde un periférico en RAM.
SAVE	$FFD8 (I)	65496	Graba directamente en un periférico desde RAM.

Las presentes tablas contienen TODAS las rutinas KERNAL, algunas de las cuales no están disponibles en todos los modelos de CPU por no disponer éstos de tantas posibilidades (RS232C, por ejemplo).

Por otro lado, el Kernal emite, independientemente de sus equivalentes BASIC, un cierto número de mensajes de error internos. Dichos mensajes serán posteriormente tratados por los lenguajes y comunicados en su caso al operador. Recordad que en el caso de Commodore, el lenguaje BASIC se encuentra siempre residente junto al Kernal, actuando de modo íntimamente relacionado con él.

En su versión de base (en los equipos actuales, cuando menos) Commodore no prevé gestiones de ficheros de claves, tablas ni tratamiento de pantallas especiales (aunque posee un avanzado y fácilmente comprensible manejo de ficheros y un excelente editor de pantalla en sus sistemas). Es por ello que el departamento de software MEC.SOFT, ha desarrollado un sistema operativo coresidente con el Kernal y un cointerpretador BASIC para el mismo, llamados MEC/DOS en su conjunto.

* * *

Nota: Las rutinas de la tabla 1.7, rutinas de canal, también se aplican al bus IEEE.

JUEGOS

Correlaciones



por **DANIEL GIMÉNEZ CAPILLA**

Dentro del trabajo estadístico, cada día más necesario, las correlaciones ocupan un campo muy importante. Y el tratamiento informático permite que lleguen a él personas poco prácticas en estadística.

Voy a explicarlo, referido al campo de correlaciones, con dos ejemplos, para que cualquier «VICadicto» pueda manejarlo a su antojo y obtener buenas prestaciones. Se tratará de ver en cada ejemplo si dos dimensiones guardan alguna relación de dependencia entre sí, y si esa relación está en proporción directa o inversa.

1. «Supongamos que se decide hacer un curso para mandos intermedios en una Empresa, a base de clases teóricas(X) y prácticas(Y), y que la gerencia desea saber si los programas de los cursos estaban bien montados. Los promedios obtenidos en las clases son:

(X)TEORIA 3-5-4-7-8-3-4-2-6-8-9-6-5-5
(Y)PRACTI 4-2-7-6-8-5-5-3-7-9-5-8-5-2

Total: 14 asistentes a clase.

Entramos el programa. Respondemos los INPUT en la forma: «Valor 1 de X=3/valor 1 de Y=4/valor 2 de X=5/valor 2 de Y=2/etc. El VIC nos dice inmediatamente después del RETURN final:

COEF. CORRELACION: 0.52
MINIMA CORRELACION

La conclusión está a la vista: alguna cosa ha fallado en el sistema de formación; se está gastando dinero en algo que no cubre bien los objetivos propuestos.»

2. «En una Empresa se encarga a dos examinadores seleccionar a un ejecutivo. Van a valorar: (A)ESTUDIOS, (B)PRESENCIA, (C)ATRACTIVO, (D)EXPERIENCIA, (E)DECISION y (F) MADUREZ PERSONAL. Una vez valorado, las puntuaciones de cada juez fueron:

Aspecto	Juez X	Juez Y
A	2	1
B	1	2
C	4	5
D	3	6
E	6	4
F	5	3

Cuando las valoraciones pasaron por el VIC, el jefe de Empresa decidió enviar a los dos examinadores a paseo y entrevistar personalmente al candidato, ¿por qué?

Entremos el programa y veamos si había correlación entre ambas apreciaciones.

* * * * * *

El programa puede simplificarse, pero lo he hecho en forma didáctica para que se vea más clarito, por si hay alguien a quien interesa aprender estadística de paso. De todas maneras, hay que andarse con ojo con las dimensiones para que el VIC no nos sorprenda con su aterrador OUT OF MEMORY. ■

LISTADO

```
1 PRINT"[ CLR ]***CORRELACIONES***"

10 X=0:Y=0:P=0:Q=0:J=0
15 REM ENTRAR DATOS
20 INPUT"CASOS=";N
25 DIMA(N),B(N),C(N),D(N),X(N),Y(N)
,L(N)
30 FORI=1TON
40 PRINT"VALOR"I"DE X=":INPUTX(I)
50 PRINT"VALOR"I"DE Y=":INPUTY(I)
60 NEXTI
62 REM PROMEDIOS
70 FORI=1TON
80 X=X+X(I):Y=Y+Y(I)
90 NEXTI
100 M=X/N:T=Y/N
120 REM DESVIACIONES
210 FORI=1TON
220 A(I)=X(I)-M
221 B(I)=Y(I)-T
222 C(I)=A(I)↑2
223 D(I)=B(I)↑2
224 NEXTI
228 FORI=1TON
229 P=P+C(I)
230 Q=Q+D(I)
231 NEXTI
235 FORI=1TON
236 L(I)=A(I)*B(I)
237 J=J+L(I)
238 NEXTI
240 FORI=1TON
244 K=SQR(P):F=SQR(Q):R=J/(K*F)
245 PRINT"[ CLR ]"
246 PRINT"COEF.CORRELACION="R
247 FORA=1TO7:PRINT:NEXTA
249 REM EFICACIA
250 IFR>1THENPRINT"ERROR DE CALCULO
"
251 IFR<.5THENPRINT"NO HAY CORRELAC
ION"
252 IFR=>.5ANDR=<.6THENPRINT"MINIMA
 CORRELACION"
253 IFR<.6ANDR>.7THENPRINT"CORRELAC
ION ACEPTABLE"
254 IFR=>.7ANDR<.81THENPRINT"ALTA C
ORRELACION"
255 IFR=>.81ANDR<1THENPRINT"MUY ALT
A CORRELACION"
256 IFR=1THENPRINT"CORRELACION PERF
ECTA"
257 END
READY.
```



JUEGOS

Bombardeo

por **C. PÉREZ RODRÍGUEZ**

Me llamo Clemente Pérez Rodríguez, vivo en Zaragoza, tengo 22 años y estoy suscrito a «Club Commodore», revista que encuentro muy interesante aunque es cierto que faltan colaboraciones.

Por eso, queriendo aportar mi grano de arena, os envío un programa que he creado. Se trata de un juego que me parece original aunque haya otros parecidos.

Consiste en lo siguiente:

Un bombardero sobrevuela una casita y trata de alcanzarla con una bomba. La bomba se suelta pulsando la letra «B» pero no cae verticalmente sino que describe una trayectoria parabólica que depende de la velocidad del avión y de la altura a la que se encuentra.

La velocidad es aleatoria y el avión comienza volando por la parte alta de la pantalla. Cada vez que cubre una línea, baja un poco, siendo más fácil hacer blanco. Pero si bajamos demasiado perderemos la oportunidad de disparar.

El número de veces que se puede bajar depende de la velocidad y el azar.

Disponemos de tres bombas pero cada vez que se hace blanco son restituidas. El tanteo por hacer blanco depende de la altura, la velocidad y de si hemos acertado a la casita de lleno.

El programa tiene sonido, colores y caracteres propios. Ha de ejecutarse sin expansión de memoria.

¡Buena suerte!

La estructura del programa es la siguiente:

- 100-230: Instrucciones y creación de caracteres
- 240: Bombas y tanteo iniciales
- 250-300: Dibuja el blanco y da la velocidad del avión
- 310-320: Imprime el tanteo
- 330-410: Vuelo del avión esperando a que se pulse «b»
- 420-430: Imprime «LENTO» si se ha perdido la oportunidad de disparar
- 440-600: Trayectoria de la bomba y del avión
- 610-700: Dibuja y sonoriza la explosión y comprueba si ha habido blanco
- 710-730: Final de partida
- 740-780: Subrutina silbido de la bomba
- 790-830: Subrutina ruido de explosión
- 840-880: Data para generar caracteres

Algunas de las variables que se utilizan:

- O - Variable de ayuda en la trayectoria de la bomba
- C - Situación del blanco
- E - Error en el blanco
- XI, WI - Coordenadas de la bomba
- G - Situación de la bomba
- X,Y - Coordenadas del avión
- K - Situación del avión
- S, S$ - Puntuación
- YI - Altura desde la que se lanzó la bomba
- V - Velocidad del avión
- B - Número de bombas disponibles

LISTADO

```
100 PRINT"UN MOMENTO, POR FAVOR"
110 A=7168:FORI=ATOA+511:POKEI,PEEK
(I+25600):NEXT
120 READN:IFNTHENFORN=NTON+15:READA
:POKEN,A:NEXT:GOTO120
130 POKE36879,59
140 POKE56,28:POKE52,28:POKE55,0:PO
KE36869,255
150 PRINT"[ CLR ][ BLK ][RVSON][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRR][CRSRR]
CLEMENTE PEREZ R.[RVSOF]"
160 PRINT"[RVSON][CRSRD][CRSRD][CRSRD]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[CRSRR]PRESENTA:[RVSOF]"
170 GOSUB740
180 PRINT"[RVSON][ CLR ][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRD][CRSRD][CRSRD]
[CRSRD]EL BOMBARDERO[RVSOF]"
190 PRINT"[RVSON][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRR]BOMBARD
EA LA CASITA[RVSOF]"
200 PRINT"[RVSON][CRSRD][CRSRR]USAN
DO LA LETRA /B/[RVSOF]"
210 GOSUB790
220 PRINT"[ CLR ]"
230 POKE36869,255
240 B=3:S=0
250 FORN=8164TO8185:POKEN,5:POKEN+3
0720,5:NEXT
260 FORN=8164TO8163+B:POKEN,4:POKEN
+30720,4:NEXT
270 C=INT(RND(1)*11)+8175
280 POKEC,2:POKEC+30720,2
290 POKE36878,10:POKE36877,150
300 V=INT(RND(1)*3)+1:IFV=3THENV=4
310 S$="      "+STR$(S):S$=RIGHT$(S$
,6)
320 PRINT"[HOME ][CRSRR][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[RVSON]";S$;"[RVSOF]"
330 POKE198,0
340 FORY=1TO3*(V+INT(RND(1)*3))STEP
3
350 FORX=0TO20
360 K=7680+Y*22+X
370 POKEK+30720,0:POKEK+30721,0:POK
EK,0:POKEK+1,1
380 FORT=1TO80/V:NEXT
390 GETA$:IFA$="B"THEN440
400 POKEK,160:POKEK+1,160
410 NEXTX,Y
420 PRINT"[HOME ][RVSON][ BLK ][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRR]LENTO![RVSOF]"

430 FORT=1TO1000:NEXT:GOTO710
440 YI=Y+1:XI=X:K=K-X
450 L=230
460 FORWI=YITO21
470 IFX<21THENPOKEK+X+30721,0:POKEK
+X,0:POKEK+X+1,1:GOTO490
480 POKE36877,0
490 L=L-1:POKE36876,L
500 O=INT((V+2)*SQR(21-WI)/12)
510 XI=XI+O:IFXI>21THEN570
520 G=7680+22*WI+XI
530 L=L-1:POKE36876,L
540 POKEG+30720,4:POKEG,4
550 FORT=1TO19:NEXT
560 L=L-1:POKE36876,L
570 POKEG,160
580 POKEK+X,160:POKEK+X+1,32:X=X+1
590 L=L-1:POKE36876,L
600 NEXT
610 G=8142+XI:IFG<8164THEN640
620 IFG=8164THENPOKE38905,1:POKE818
5,6
630 GOTO660
640 POKEG+30720,1:POKEG+30741,1:POK
EG+30742,1:POKEG+30743,1
650 POKEG,3:POKEG+21,6:POKEG+22,7:P
OKEG+23,6
660 GOSUB790
670 IFPEEK(C)=2THEN710
680 E=0:IFPEEK(C)=6THENE=5
690 B=3:S=S+(27-YI)*V-E
700 PRINT"[ CLR ]":GOTO250
710 B=B-1:IFB=0THENPRINT"[HOME ][RVSON]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[CRSRR][CRSRR]FIN[CRSRD][CRSRD][CRSRD]
[CRSRL][CRSRL][CRSRL][CRSRL][CRSRL]
[CRSRL]PULSA CTRL[RVSOF]":POKE36877
,0:GOTO730
720 PRINT"[ CLR ]":GOTO250
730 POKE198,0:WAIT653,4:GOTO150
740 POKE36878,10
750 FORL=230TO150STEP-1:POKE36876,L

760 FORM=1TO16:NEXTM,L
770 POKE36876,0:POKE36878,0
780 RETURN
790 POKE36877,220:POKE36876,0
800 FORL=15TO0STEP-1:POKE36878,L
810 FORM=1TO250:NEXTM,L
820 POKE36877,0:POKE36878,0
830 RETURN
840 DATA 7168 ,193,226,255,255,255,
255,0,0,242,170,254,255,254,250,98,
0
850 DATA 7200 ,0,40,56,56,16;16,0,0
,0,0,0,0,0,0.0,255
860 DATA 7184 ,24,60,126,255,66,90,
90,126,24,94,255,127,255,118,60,24
870 DATA 7216 ,0,24,56,30,191,255,2
55,255,60,90,60,24,26,219,255,255
880 DATA 0
READY.
```

COLABORACIONES

dibujo en perspectiva ("dibuvic")

por **ALFONSO IZQUIERDO FONT**

Este programa permite la representación en pantalla de una figura en perspectiva.

La forma de hacerlo es mediante una matriz (matriz de proyección, PRO en el listado).

Esta matriz de tres elementos por dos, es una función proyección del espacio en el plano.

Bueno, en resumen, que teniendo las coordenadas de los puntos, aplicándoles esta matriz, efectuar su producto, nos resulta un punto en el plano, con dos coordenadas.

Esta matriz es la que resulta de todo el complejo aparato matemático de Euler, siendo función de unos ángulos, ángulos de visión. Tres ángulos

LISTADO

```
30 GRAPHIC0:GOSUB5000:PRINT"[ CLR ]
    MENU[CRSRD][CRSRL][CRSRL][CRSRL]
[CRSRL]----"
32 AL=-.78:BE=.78:GA=-.78
34 CX=500:CY=500:ES=10
40 PRINT"[CRSRD]  1 END"
50 PRINT"[CRSRD]  2 CREAR DATOS"
60 PRINT"[CRSRD]  3 DIBUJAR"
70 PRINT"[CRSRD]  4 CREAR FIC.CINTA
"
80 PRINT"[CRSRD]  5 LEER FIC.CINTA"

90 INPUT"[CRSRD][CRSRD]OPCION ";A
100 ON A GOTO 690,110,220,580,640,1
500
110 CLR:AL=-.78:BE=.78:GA=-.78
114 CX=500:CY=500:ES=10
118 PRINT"[ CLR ][CRSRD][CRSRD][CRSRD]
":INPUT"NO. DE PUNTOS";N:DIMA(N,3),
B$(N),PRO(3,2)
120 FORI=1TON:FORJ=1TO3:PRINTI;"
 A("I","J")=";:INPUTA(I,J):NEXT:NEX
T
130 PRINT"PUNTO","COOR."
140 PRINT"-----","-----"
150 K=N:R=1:IFK>20THENK=20
160 FORI=RTOK:PRINTI;"  ("A(I,1)","
A(I,2)","A(I,3)")":NEXT:GOSUB10000:
IFI>NTHEN180
170 IFN>20THENR=N-20:K=N:GOTO160
180 PRINT"[ CLR ][CRSRD][CRSRD]DAME
 LA LISTA DE CONEXION ENTRE PUNTOS
EJ.(0512) QUE SERA EL I CONECTADO C
ON"
190 PRINT" EL 05 Y TAMBIEN CON  EL
12)
200 FORI=1TON:PRINT"NUDO "I;:INPUTB
$(I):NEXT
210 GOTO30
220 GRAPHIC0:PRINT"[CRSRD] FUNCIONA
MIENTO[CRSRD][CRSRL][CRSRL][CRSRL][CRSRL]
[CRSRL][CRSRL][CRSRL][CRSRL][CRSRL]
[CRSRL][CRSRL][CRSRL][CRSRL][CRSRL]
--------------"
230 PRINT"[CRSRD] A-S GIRA SEGUN XX
"
240 PRINT"[CRSRD] B-N GIRA SEGUN YY
"
250 PRINT"[CRSRD] G-H GIRA SEGUN ZZ
"
260 PRINT"[CRSRD]  +  AUMENTA ESCAL
A"
270 PRINT"[CRSRD]  -  DISMINUYE ESC
ALA"
273 PRINT"[CRSRD]  1  INSTRUCCIONES
"
275 PRINT"[CRSRD]  /  RUN MENU"
280 PRINT"[CRSRD][CRSRD]PULSE [ BLU ]
[RVSON]←[RVSOF]"
290 GETA$:IFA$=""THEN290
300 GRAPHIC2:COLOR0,0,1,1
320 PRO(1,1)=(SIN(GA)*COS(AL)-COS(G
A)*SIN(BE)*SIN(AL))*ES
330 PRO(1,2)=(COS(GA)*SIN(BE)*COS(A
L)+SIN(GA)*SIN(AL))*ES
340 PRO(2,1)=(COS(GA)*COS(AL)+SIN(G
A)*SIN(BE)*SIN(AL))*ES
350 PRO(2,2)=(COS(GA)*SIN(AL)-SIN(G
A)*SIN(BE)*COS(AL))*ES
360 PRO(3,1)=(-SIN(AL)*COS(BE))*ES
370 PRO(3,2)=COS(AL)*COS(BE)*ES
380 FORI=1TON:R=VAL(RIGHT$(B$(I),2)
):S=VAL(LEFT$(B$(I),2))
385 X1=A(I,1)*PRO(1,1)+A(I,2)*PRO(2
,1)+A(I,3)*PRO(3,1)+CX
390 Y1=A(I,1)*PRO(1,2)+A(I,2)*PRO(2
,2)+A(I,3)*PRO(3,2)+CY
400 X2=A(R,1)*PRO(1,1)+A(R,2)*PRO(2
,1)+A(R,3)*PRO(3,1)+CX
410 Y2=A(R,1)*PRO(1,2)+A(R,2)*PRO(2
,2)+A(R,3)*PRO(3,2)+CY
420 X3=A(S,1)*PRO(1,1)+A(S,2)*PRO(2
,1)+A(S,3)*PRO(3,1)+CX
430 Y3=A(S,1)*PRO(1,2)+A(S,2)*PRO(2
,2)+A(S,3)*PRO(3,2)+CY
440 DRAW2,X1,Y1 TO X2,Y2:DRAW2,X1,Y
1 TO X3,Y3:NEXT
450 CHAR19,0,"?"
460 GETA$:IFA$=""THEN460
470 IFA$="A"THENAL=AL+.2
480 IFA$="S"THENAL=AL-.2
490 IFA$="B"THENBE=BE+.2
500 IFA$="N"THENBE=BE-.2
510 IFA$="G"THENGA=GA+.2
520 IFA$="H"THENGA=GA-.2
530 IFA$="+"THENES=ES+5
540 IFA$="-"THENES=ES-5
542 IFA$="1"THEN 220
550 IFA$="/"THEN 30
560 SCNCLR:GOTO320
580 PRINT"[ CLR ] COLOCA LA CINTA L
ISTA PARA GRABAR"
590 PRINT"[CRSRD][CRSRD]NOMBRE DEL
ARCHIVO":INPUTN$
600 OPEN1,1,1,N$:PRINT#1,N
610 FORI=1TON:FORJ=1TO3:PRINT#1,A(I
,J):NEXT:NEXT
620 FORI=1TON:PRINT#1,B$(I):NEXT
630 CLOSE1
635 GOTO30
640 CLR:PRINT"[ CLR ][CRSRD][CRSRD]
NOMBRE DEL ARCHIVO":INPUTN$
650 OPEN1,1,0,N$:INPUT#1,N
660 FORI=1TON:FORJ=1TO3:INPUT#1,A(I
,J):NEXT:NEXT
670 FORI=1TON:INPUT#1,B$(I):NEXT
680 CLOSE1
685 GOTO30
690 END
5000 POKE36864,12:POKE36865,38:RETU
RN
10000 PRINT"SON CORRECTOS ESTOS   P
UNTOS(S/N)":INPUTA$
10010 IFA$<>"N"THENRETURN
10020 PRINT"[CRSRU][CRSRU]DA EL PUN
TO INCORRECTO (FIN CON 0)"
10030 INPUT"NO. Y COOR.";Z,A,B,C
10040 IFZ<>0THENA(Z,1)=A:A(Z,2)=B:A
(Z,3)=C:GOTO10030
10050 RETURN
READY.
```

que determinan la forma de divisar el objeto.

NO ES NECESARIO ENTRAR PARÁMETROS DE DIBUJO

El programa trabaja en perspectiva isométrica y no es necesario entrar parámetros de dibujo.

Los únicos parámetros que debemos entrar son los puntos que tendrá la figura y la conexión entre ellos ya que el programa no dispone de una forma fácil la disposición del dibujo.

EL PROGRAMA OFRECE DISTINTAS OPCIONES

Dentro de todo esto, el programa tiene una serie de opciones bastante interesantes, desde la de entrar la figura hasta la de modificar las coordenadas de un punto de ella, pasando por el cambio de punto de vista de la figura, paseándola por toda la pantalla, modificando la escala, y variando los ángulos de vista.

MANEJO DEL PROGRAMA

El manejo del programa es sencillo.

Lo primero es entrar los datos. Esto se efectúa dando el número de puntos que posee la figura.

Después, debemos entrar las coordenadas de todos los puntos, teniendo en cuenta que la numeración que nos presenta el ordenador por pantalla es la que utilizaremos para la conexión entre puntos.

Cuando hemos finalizado esta tarea, podemos corregir algún dato erróneo que hayamos entrado.

La entrada de datos para la conexión es igual de sencillo:

Debemos entrar cuatro caracteres por número presentado en pantalla.

Estos cuatro caracteres son dos para el primer punto conectado, y los dos siguientes para el segundo.

Por ejemplo, si entramos un tetraedro, entraríamos:

```
1   0 return
    0  "
    0  "
2   1  "
    0  "
    0  "
3   0  "
    0  "
    1  "
4   0  "
    1  "
    0  "
```

Y después deberíamos entrar la conexión entre puntos:

```
1  0203
2  0304
3  0401
4  0102
```

Esto aunque parezca insuficiente, no lo es ya que podemos entrar todas las conexiones y además, nos sobrará.

Si sucede esto, sólo tenemos que hacer que el ordenador nos dibuje dos veces una misma línea repitiendo, si es necesario, alguna conexión.

Cuando hemos acabado todo esto, es conveniente grabar los datos, y proceder a dibujar.

Para dibujar, sólo tendremos que pulsar el 3 en el menú y nos aparecerá una explicación del funcionamiento interactivo del programa.

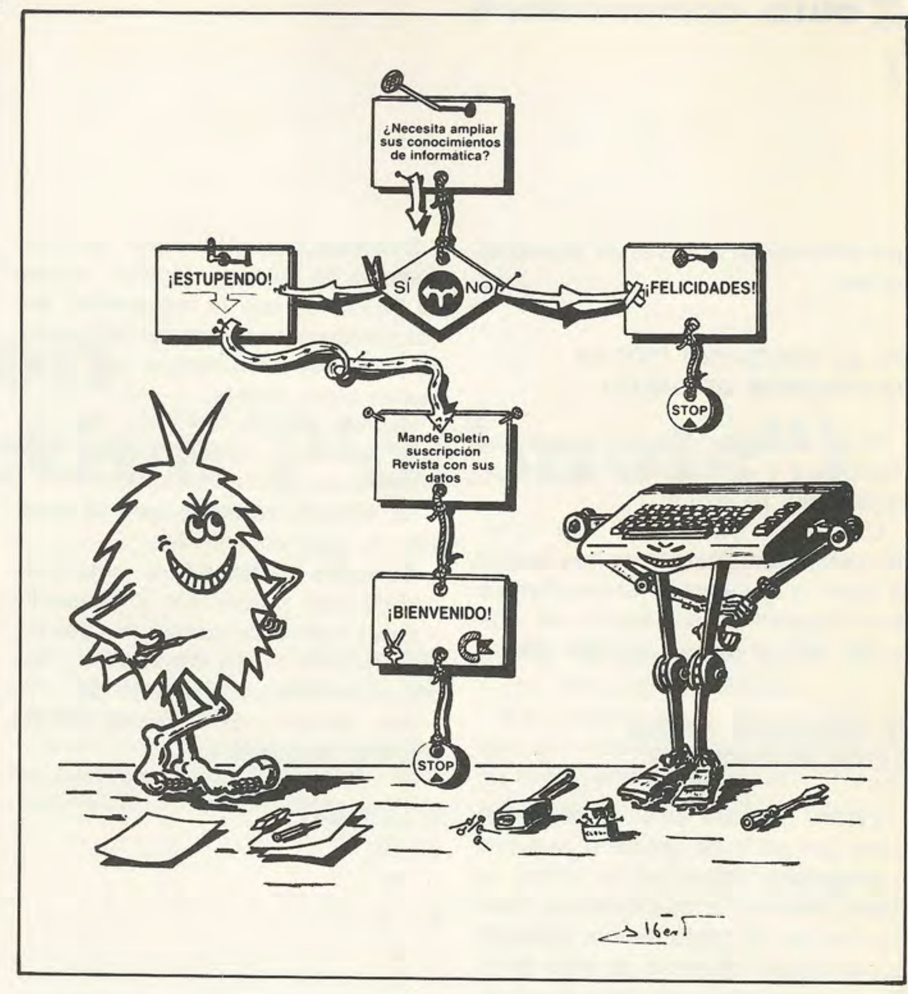
Del resto del programa huelgan comentarios, ya que son apartados que se explican solos en él.

Es fácil que este programa sirva para la docencia, para la visualización de figuras, cosa que cuesta mucho a los estudiantes.

DIFERENTES POSIBILIDADES DE DIBUJO

También es de destacar, la posibilidad de dejar la figura en planta, alzado o perfil, lo que también es muy interesante. ■

clave para interpretar los listados de CLUB COMMODORE



Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD] = Tecla cursor hacia abajo (sin SHIFT)
[CRSRU] = Tecla cursor hacia arriba (con SHIFT)
[CRSRR] = Tecla cursor a la derecha (sin SHIFT)
[CRSRL] = Tecla cursor a la izquierda (con SHIFT)
[HOME] = Tecla CLR/HOME (sin SHIFT)
[CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOF] respecto a la tecla CTRL y las teclas 9 y 0.

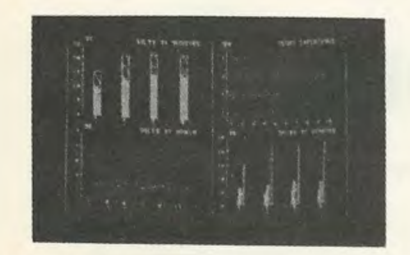
El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo — por ejemplo [COMM+] o [SHIFA] —. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente. ■



SAKATI, S. A. SOFTWARE PROFESIONAL

Ardemans, 24 - Madrid-28 - Tel. 2567794 - Telex 44222 CICI E

¿ES UD. USUARIO DE UN COMMODORE-64...? SI ES ASÍ ENTONCES NUESTROS PROGRAMAS SON PARA UD.



GRAFICOS-64

Le permite trabajar en modo alta resolución. Traza puntos, líneas, rellena áreas de color, define colores de pantalla y fondo, abre ventanas en la pantalla, escribe y superpone textos en los gráficos, invierte textos. 35 nuevos comandos harán de su trabajo una diversión. MANUAL EN CASTELLANO.

P.V.P. en disco 9.000.— Pts.
P.V.P. en cinta 7.500.— Pts.

EXTENSION DE BASIC

Potencie el sistema operativo de su COMMODORE-64 con este cartucho que añade 42 nuevos comandos - de gran utilidad. Trabaje con BASIC 4.0 de COMMODORE. Podrá localidar y corregir programas con gran facilidad, pasar programas de un CBM-8000 a un 64, potenciar el editor, y conectar sin ningún interface, sólo con un cable, su 64 a una impresora CENTRONIC. MANUAL EN CASTELLANO.

P.V.P. 25.000.— Pts.

DESENSAMBLADOR

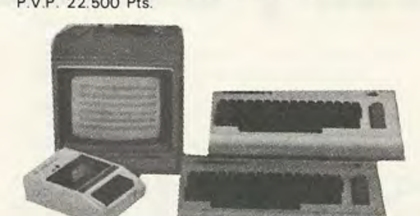
Le permite desensamblar cualquier programa realizado en código máquina. INSTRUCCIONES EN CASTELLANO.

P.V.P. 2.000 Pts. en cinta.

BASE DE DATOS I

Es algo más que un potente generador de aplicaciones. Los programas por aplicación son ilimitados. Aporta un nuevo lenguaje con más de 40 nuevos comandos. Podrá diseñar su propia configuración, con registros de hasta 1000 caracteres y gran facilidad de modificación de los mismos. 127 campos por registro. Contará con 15 ficheros por aplicación ilimitados. Los ficheros de clave y de memoria son así mismo ilimitados. Enlace con EASY SCRIP. Calendario aritmético. Esta Base de Datos es el número uno en su género, algo que Vd. debe tener para su trabajo. MANUAL EN CASTELLANO.

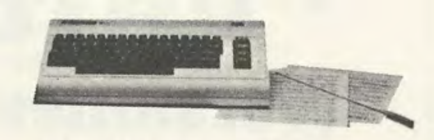
P.V.P. 22.500 Pts.



BASE DE DATOS II

12 opciones presentadas en el menú. Cree nuevos ficheros de datos. Cuenta Vd. con 12 campos. Esta Base de Datos es ideal para personas que empiezan a introducirse en este método de trabajo

P.V.P. 4.000 Pts.



¿LE GUSTARIA COMPONER MUSICA?

Nuestro programa hará de Vd. un melómano. MUSICA-64 controla tres voces en un rango de 8 octavas cada voz. Sonarán en cualquiera de estos 4 instrumentos: órgano, clavicornio, silofón y trompeta. Podrá introducir 2.000 notas por voz, y una vez compuesta su música, archívela en cinta o en disco. INSTRUCCIONES EN CASTELLANO.

P.V.P. en cinta 4.000 Pts.
P.V.P. en disco 5.000 Pts.

QUIMICA

Lecciones asequibles de química para sus hijos y para todos los padres de los hijos que estudian química.

P.V.P. 2.000 Pts. en cinta

VELOS AK-64

Su soporte para programar, capaz de ordenar 500 matrices en 30 segundos.

P.V.P. 2.000 Pts. en cinta

¡NO NOS OLVIDAMOS DE LOS NIÑOS NI DE LA TARDE DEL DOMINGO!

JUEGO DE LOS PALILLOS

Cace los palillos movibles y con joy-stick enciérrelos en el menor área posible, pero ¡ojo! los palillos se mueven en todas direcciones

GUERRA DE TANQUES

Prepárese a rechazar un ataque de la brigada de tanques demostrando su puntería.

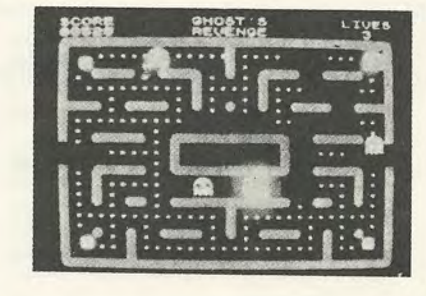
CACTUS

En su desierto particular, tiene Vd. un hermoso cactus al que amenazan los topos, las águilas y los pájaros.

COMECOCOS

El popular juego ahora disponible para su 64.

P.V.P. 2.000 Pts. por juego en cinta
P.V.P. 6.000 Pts. por los 4 juegos en disco



BOLETIN DE PEDIDO

enviar a: **SAKATI, S. A.** Ardemans, 24. MADRID-28

Ref.		Cdad.	Precio
		TOTAL	

Talon adjunto ☐ Contra Reembolso

echa: Firma:

º Calle

ódigo Postal Ciudad

ovincia Teléfono

¡TENEMOS ALGO MUY IMPORTANTE! CONTROL DE ALMACEN

Toda aquella empresa que tenga problemas de almacén no debe pararse en este anuncio. Debe llamarnos y le explicaremos con detalle este programa. El espacio no da para más, pero he aquí un avance:

- 1.500 referencias son controladas por almacén, en unidades, litros, kilos, metros, etc., stock mínimo, y lo propio de estos programas, etiquetas, etc.

  Pero además el programa tiene una opción de anotación para los gratos de su empresa, con análisis de rentabilidad de su negocio, sin pasar por contabilidad. De manejo sencillo. TODO EN CASTELLANO.

  El programa se presenta en disco.

P.V.P. 35.000 Pts. Ref: SD14

JUEGOS

"saltador" (versiones VIC y COMMODORE 64)

por **R. PARDO**

Este juego consiste en hacer bajar al saltador por las escaleras sin que caiga por unos agujeros abiertos arteramente por el ordenador. Entre la versión de programa para el VIC con la figuración de memoria mínima (es decir, sin expansión alguna). En la versión del 64, no hay ningún problema.

Una vez funcione, en la pantalla del VIC aparecerán unas estructuras unidas por unas escaleras; después de unos instantes saldrá nuestro «saltador» caminando normalmente. Cuando se aproxime a un agujero deberá hacerlo saltar. Si lo hace demasiado pronto, es muy probable que el saltador caiga por el agujero. Si llega a bajar por la última escalera, el ordenador cambiará de pantalla, y acelerará automáticamente al saltador (sólo en la versión del VIC). En la versión del 64 al inicio del programa muestra las instrucciones de juego.

Para hacer saltar el saltador debe pulsar la barra espaciadora en ambas versiones, o el botón de fuego del joystick en la versión del 64.

En ambas versiones se incrementa el tanteo medio al cambiar de pantalla.

LISTADO PARA LA VERSIÓN DEL VIC

```
0 M=3:T=150:D=5:X=25:P=61:POKE55,16
0:POKE56,29:S=36876:POKE36878,15:GO
TO10012

1 C=27:F%=5:FORI=7680TO8185:POKEI,5
9:NEXT

2 FORI=7702TO7723:POKEI,53:NEXT:FOR
I=7812TO7833:POKEI,53:NEXT:FORI=790
0TO7921:POKEI,53:NEXT

3 POKE36879,C:FORI=8032TO8053:POKEI
,53:NEXT:FORI=8142TO8163:POKEI,53:N
EXT

4 FORI=38400TO38884+21:POKEI,4:NEXT

5 FORI=38488TO38510+21:POKEI,F%:NEX
T

6 FORI=38576TO38598+21:POKEI,F%:NEX
T

7 FORI=38708TO38730+21:POKEI,F%:NEX
T

8 FORI=38818TO38840+21:POKEI,F%:NEX
T:GOSUB10020:FORI=1TO1000:NEXT

9 I=7790

10 IFI/2=INT(I/2)THENPOKEI-1,59:GOS
UB110

11 IFI/2=INT(I/2)THENPOKEI,55:FORJ=
1TOT:NEXT:GOTO14

13 POKEI-1,59:POKEI,56:FORJ=1TOT:NE
XT:B=7812:GOSUB510

14 IFPEEK(197)=32THENGOSUB20

15 IFPEEK(I+22)=54THENPOKEI,59:GOTO
30

16 IFPEEK(I+22)=60THEN500

17 I=I+1:IFI>7811THENI=7790:POKE781
1,59

18 GOTO10

20 I=I-21:POKEI+21,59

21 IFPEEK(I+22)<>59ORPEEK(I+44)<>53
THENSC=SC+X:POKEI-22,P:GOSUB112:POK
EI-22,59

23 POKEI,55:FORJ=1TOT:NEXT:I=I+23:I
FI>7811THENI=7790:POKE7811,59

24 POKE7789,59:POKE7790,59

25 FORJ=1TOT:NEXT:POKEI-23,59:POKEI
,55:RETURN

30 I=7898

31 IFI/2=INT(I/2)THENPOKEI+1,59:GOS
UB110

32 IFI/2=INT(I/2)THENPOKEI,58:FORJ=
1TOT:NEXT:GOTO34

33 POKEI+1,59:POKEI,57:FORJ=1TOT:NE
XT:B=7900:GOSUB510

34 IFPEEK(197)=32THENGOSUB40

35 IFPEEK(I+22)=54THENPOKEI,59:GOTO
50

36 IFPEEK(I+22)=60THEN500

37 I=I-1:IFI<7878THENI=7898:POKE787
8,59

38 GOTO31

40 I=I-23:POKEI+23,59

41 IFPEEK(I+22)<>59ORPEEK(I+44)<>53
THENSC=SC+X:POKEI-22,P:GOSUB112:POK
EI-22,59

43 POKEI,58:FORJ=1TOT:NEXT:I=I+21:I
FI<7878THENI=7898:POKE7878,59

44 POKE7856,59:POKE7855,59

45 FORJ=1TOT:NEXT:POKEI-21,59:POKEI
,58:RETURN

50 I=8010

51 IFI/2=INT(I/2)THENPOKEI-1,59:GOS
UB110

52 IFI/2=INT(I/2)THENPOKEI,55:FORJ=
1TOT:NEXT:GOTO54

53 POKEI-1,59:POKEI,56:FORJ=1TOT:NE
XT:B=8032:GOSUB510

54 IFPEEK(197)=32THENGOSUB60

55 IFPEEK(I+22)=54THENPOKEI,59:GOTO
70

56 IFPEEK(I+22)=60THEN500

57 I=I+1:IFI>8031THENI=8010:POKE803
1,59

58 GOTO51

60 I=I-21:POKEI+21,59:IFPEEK(I)<>59
THENSC=SC+300

61 IFPEEK(I+22)<>59ORPEEK(I+44)<>53
THENSC=SC+X:POKEI-22,P:GOSUB112:POK
EI-22,59

63 POKEI,55:FORJ=1TOT:NEXT:I=I+23:I
FI>8031THENI=8010:POKE8031,59

64 POKE8009,59:POKE8010,59

65 FORJ=1TOT:NEXT:POKEI-23,59:POKEI
,55:RETURN
```

(continúa en la pág. siguiente)

LISTADO PARA LA VERSIÓN DEL VIC

```
70 I=8140

71 IFI/2=INT(I/2)THENPOKEI+1,59:GOS
UB110

72 IFI/2=INT(I/2)THENPOKEI,58:FORJ=
1TOT:NEXT:GOTO74

73 POKEI+1,59:POKEI,57:FORJ=1TOT:NE
XT:B=8142:GOSUB510

74 IFPEEK(197)=32THENGOSUB80

75 IFPEEK(I+22)=54THENPOKEI,59:GOTO
100

76 IFPEEK(I+22)=60THEN500

77 I=I-1:IFI<8120THENI=8140:POKE812
0,59

78 GOTO71

80 I=I-23:POKEI+23,59

81 IFPEEK(I+22)<>59ORPEEK(I+44)<>53
THENSC=SC+X:POKEI-22,P:GOSUB112:POK
EI-22,59

83 POKEI,58:FORJ=1TOT:NEXT:I=I+21:I
FI<8120THENI=8140:POKE8120,59

84 POKE8098,59:POKE8097,59

85 FORJ=1TOT:NEXT:POKEI-21,59:POKEI
,58:RETURN

100 P=P+1:IFP=64THENP=61

101 D=D-1:T=T-50

102 X=X+50:IFX>125THENX=25:D=8:T=15
0:C=27:F%=5

103 IFX=75THENC=232:F%=0

104 IFX=125THENC=8:F%=7

105 GOTO2

110 POKES,140:FORY=1TO10:NEXT:POKES
,0:RETURN

111 POKES+1,190:FORY=1TO25:NEXT:POK
ES+1,0:RETURN

112 FORO=1TO15:POKES,200+O:NEXTO:PO
KES,0:RETURN

113 FORO=20TO0STEP-1:POKES,230+O:FO
RY=1TO25:NEXTY,O:POKES,0:RETURN

500 GOSUB113:M=M-1:IFM=0THEN502

501 P=61:X=25:D=6:C=27:T=150:F%=5:P
OKEI,59:GOTO2

502 POKE36869,240:PRINTCHR$(147);SP
C(225);"SE ACABO!":PRINT:PRINT"SU P
UNTUACION ES ";SC

503 PRINT:PRINT"OTRA VEZ?"

504 K=PEEK(197):IFK=32ORK=64THEN504

505 IFK=41THENRUN

506 END

510 IFINT(RND(1)*D)+1<>1THENRETURN

511 L=INT(RND(1)*21)+1:IFL=20ORL=1T
HEN511

512 POKEB+L,60:GOSUB111:RETURN

10000 DATA255,129,66,66,36,36,24,25
5

10002 DATA66,126,66,66,66,126,66,66

10003 DATA12,8,13,62,44,12,18,33

10004 DATA24,16,24,24,24,16,16,24

10005 DATA24,8,24,24,24,8,8,24

10006 DATA24,8,88,62,26,24,36,66

10007 DATA0,0,0,0,0,0,0,0

10008 DATA129,66,66,66,98,34,34,34

10009 DATA27,10,27,17,27,0,0,0

10010 DATA59,10,11,9,11,0,0,0

10011 DATA91,74,91,81,91,0,0,0

10012 RESTORE:FORI=7592TO7679:READA
:POKEI,A:NEXT

10015 POKE36869,255

10016 GOTO1

10020 FORI=7832TO7898STEP22:POKEI,5
4:NEXT:FORI=7901TO8011STEP22:POKEI,
54:NEXT

10021 FORI=8052TO8140STEP22:POKEI,5
4:NEXT:FORI=38552TO38618STEP22:POKE
I,6:NEXT

10022 FORI=38621TO38731STEP22:POKEI
,6:NEXT:FORI=38772TO38860STEP22:POK
EI,6:NEXT

10023 POKE8143,54:POKE8165,54:POKE3
8863,6:POKE38885,6:RETURN

READY.
```

LISTADO PARA LA VERSIÓN DEL COMMODORE 64

```
READY.

0 REM SALTADOR PARA 64 64

5 GOSUB3000:PRINT"[ CLR ]";"[CRSRR]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
INICIALIZA"

10 M=3:T=10:D=5:X=25:P=61:POKE55,16
:POKE56,64:S=54272:POKE53281,1:GOTO
970

20 C=7:F%=5:FORI=1024TO2041:POKEI,5
9:NEXT

30 POKE53280,C:FORI=1064TO1103:POKE
I,53:NEXT:FORI=1264TO1303:POKEI,53:
NEXT

33 FORI=1424TO1463:POKEI,53:NEXT:PO
KE1425,54:POKE1702,54:POKE1865,54

40 FORI=1664TO1703:POKEI,53:NEXT:FO
RI=1864TO1903:POKEI,53:NEXT

50 FORI=55296TO56176+39:POKEI,4:NEX
T

60 FORI=55456TO55496+39:POKEI,F%:NE
XT

70 FORI=55616TO55656+39:POKEI,F%:NE
XT

80 FORI=55856TO55896+39:POKEI,F%:NE
XT

90 FORI=56056TO56096+39:POKEI,F%:NE
XT:GOSUB1000:FORI=1TO1000:NEXT

100 I=1224:POKE1302,54:POKE1425,54:
POKE1702,54:POKE1865,54

110 IFI/2=INT(I/2)THENPOKEI-1,59:GO
SUB720

120 IFI/2=INT(I/2)THENPOKEI,55:FORJ
=1TOT:NEXT:GOTO140

130 POKEI-1,59:POKEI,56:FORJ=1TOT:N
EXT:B=1264:GOSUB830

140 IFPEEK(197)=60THENGOSUB190

150 IFPEEK(I+40)=54THENPOKEI,59:GOT
O240

160 IFPEEK(I+40)=60THEN760

170 I=I+1:IFI>1263THENI=1224:POKE12
63,59

180 GOTO110

190 I=I-39:POKEI+39,59

200 IFPEEK(I+40)<>59ORPEEK(I+80)<>5
3THENSC=SC+X:POKEI-40,P:GOSUB740:PO
KEI-40,59

210 POKEI,55:FORJ=1TOT:NEXT:I=I+41:
IFI>1263THENI=1224:POKE1263,59:POKE
1223,59

220 POKE1403,59:POKE1404,59

230 FORJ=1TOT:NEXT:POKEI-41,59:POKE
I,55:RETURN

240 I=1422

250 IFI/2=INT(I/2)THENPOKEI+1,59:GO
SUB720

260 IFI/2=INT(I/2)THENPOKEI,58:FORJ
=1TOT:NEXT:GOTO280

270 POKEI+1,59:POKEI,57:FORJ=1TOT:N
EXT:B=1424:GOSUB830

280 IFPEEK(197)=60THENGOSUB330

290 IFPEEK(I+40)=54THENPOKEI,59:GOT
O380

300 IFPEEK(I+40)=60THEN760

310 I=I-1:IFI<1384THENI=1422:POKE13
84,59

320 GOTO250

330 I=I-41:POKEI+41,59

340 IFPEEK(I+40)<>59ORPEEK(I+80)<>5
3THENSC=SC+X:POKEI-40,P:GOSUB740:PO
KEI-40,59

350 POKEI,58:FORJ=1TOT:NEXT:I=I+39:
IFI<1384THENI=1422:POKE1384,59:POKE
1344,59

360 POKE1344,59:POKE1343,59

370 FORJ=1TOT:NEXT:POKEI-39,59:POKE
I,58:RETURN

380 I=1624

390 IFI/2=INT(I/2)THENPOKEI-1,59:GO
SUB720
```

(continúa en la pág. 13)

Elektrocomputer

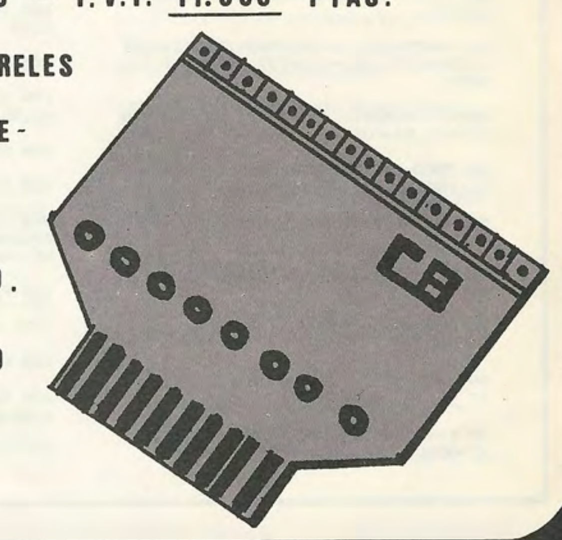
ELEKTROCOMPUTER · PRESENTA SUS NUEVOS PRODUCTOS PARA EL VIC-20 Y EL COMMODORE-64. **DATAMASTER 64** Y **CONTROLADOR · C8**, QUE AMPLIAN LAS POSIBILIDADES DE SU ORDENADOR.

DE VENTA EN DISTRIBUIDORES AUTORIZADOS DE TODA ESPAÑA.

DATAMASTER

***DATAMASTER** 64 _ SOFISTICADA BASE DE DATOS PARA EL C—64. PENSADA PARA TRABAJAR CON LA UNIDAD DE DISCO 1541. SIENDO MUY VERSATIL APROVECHA AL MAXIMO LA CAPACIDAD DE MANIOBRA Y ALMACENAMIENTO. NUMERO DE REGISTROS VARIABLE "EJ. 5000 DE 30 CARACTERES". FORMATEADOS Y COPIAS PROGRAMADAS. SALIDA A IMPRESORA (PARALELO CENTRONICS Y SERIE RS 232) CHEQUEO OPERACIONES DISCO. GARANTIA 3 MESES. MANUAL COMPL. EN CASTELLANO — P.V.P. 11.800' PTAS.

***CONTROLADOR · C8** _ CONTROLADOR DE 8 RELES PARA EL VIC-20 Y EL C-64. DE FORMA MUY SENCILLA PODEMOS HACER HASTA 255 COMBINACIONES ENTRE LOS 8 RELES, CON UN CONSUMO DE 1000 W. A 220 VOLT. CADA UNO. CON LO CUAL PODEMOS ACCIONAR TODO TIPO DE LUCES O MECANISMOS. INSTRUC. INCLUIDAS. 3 MESES GARANTIA — P.V.P. 9.800' PTAS.



VIA AUGUSTA - 120 - TEF. (93· 2180699 - BARCELONA - 6

"saltador"

(viene de la pág. 11)

LISTADO PARA LA VERSIÓN DEL COMMODORE 64

```
400 IFI/2=INT(I/2)THENPOKEI,55:FORJ
=1TOT:NEXT:GOTO420

410 POKEI-1,59:POKEI,56:FORJ=1TOT:N
EXT:B=1664:GOSUB830

420 IFPEEK(197)=60THENGOSUB470

430 IFPEEK(I+40)=54THENPOKEI,59:GOT
O520

440 IFPEEK(I+40)=60THEN760

450 I=I+1:IFI>1663THENI=1624:POKE16
63,59:POKE1623,59

460 GOTO390

470 I=I-39:POKEI+39,59:IFPEEK(I)<>5
9THENSC=SC+300

480 IFPEEK(I+40)<>59ORPEEK(I+80)<>5
3THENSC=SC+X:POKEI-40,P:GOSUB740:PO
KEI-40,59

490 POKEI,55:FORJ=1TOT:NEXT:I=I+41:
IFI>1663THENI=1624:POKE1663,59

500 POKE1641,59:POKE1624,59:POKE162
3,59

510 FORJ=1TOT:NEXT:POKEI-41,59:POKE
I,55:RETURN

520 I=1862

530 IFI/2=INT(I/2)THENPOKEI+1,59:GO
SUB720

540 IFI/2=INT(I/2)THENPOKEI,58:FORJ
=1TOT:NEXT:GOTO560

550 POKEI+1,59:POKEI,57:FORJ=1TOT:N
EXT:B=1864:GOSUB830

560 IFPEEK(197)=60THENGOSUB610

570 IFPEEK(I+40)=54THENPOKEI,59:GOT
O660

580 IFPEEK(I+40)=60THEN760

590 I=I-1:IFI<1824THENI=1862:POKE18
24,59

600 GOTO530

610 I=I-41:POKEI+41,59

620 IFPEEK(I+40)<>59ORPEEK(I+80)<>5
3THENSC=SC+X:POKEI-40,P:GOSUB740:PO
KEI-40,59

630 POKEI,58:FORJ=1TOT:NEXT:I=I+39:
IFI<1824THENI=1862:POKE1824,59

640 POKE1784,59:POKE1783,59

650 FORJ=1TOT:NEXT:POKEI-39,59:POKE
I,58:RETURN

660 P=P+1:IFP=64THENP=61

670 D=D-1:T=T-.1

680 X=X+50:IFX>125THENX=25:D=8:T=10
:C=5:F%=5

690 IFX=75THENC=0:F%=0

700 IFX=125THENC=8:F%=7

710 GOTO30

720 POKES+4,17:POKES+5,132:POKES+6,
132:POKES+24,6

721 H0=28:L0=49:POKES+1,H0:POKES,L0
:FORZ=1TO200:NEXT:GOSUB2000:RETURN

740 POKES+24,15:POKES+4,17:POKES+5,
132:POKES+6,132

741 FORH1=21TO126:POKES+1,H1:L1=181
:POKES,L1:NEXT:GOSUB2000:RETURN

760 POKES+24,15:POKES+4,17:POKES+5,
33:POKES+6,132:H2=233

765 H2=H2-5:POKES+1,H2:L2=181:POKES
,L2

766 POKEI,58:POKEI-40,59:POKEI+5427
2,0:I=I+40:IFI<1983THEN765

767 GOSUB2001

769 M=M-1:IFM=0THEN780:POKES+1,H2:L
2=181:POKES,L2:NEXT:GOSUB2001

770 P=61:X=25:D=6:C=27:T=10:F%=5:PO
KEI,59:GOTO30

780 POKE53272,21:PRINTCHR$(147);SPC
(205);"[CRSRR][CRSRR][CRSRR][CRSRR]
```

(termina en la pág. siguiente)

LISTADO PARA LA VERSIÓN DEL COMMODORE 64

```
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
SE ACABO!":PRINT

785 PRINT"[CRSRD][CRSRR][CRSRR][CRSRR]
[CRSRR][CRSRR]SU PUNTUACION ES DE";
SC

790 PRINT:PRINT"[CRSRD][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[CRSRR]OTRA VEZ?"

800 K=PEEK(197):IFK=60THEN800

810 IFK=13THENPRINT"[ CLR ]":RUN

820 IFK=39THENPRINT"[ CLR ][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRR][CRSRR]
[CRSRR][CRSRR][CRSRR][CRSRR][CRSRR]
[CRSRR]ADIOS!!";:FORW=1TO500:NEXT:P
RINT"[ CLR ]":END

825 GOTO800

830 IFINT(RND(1)*D)+1<>1THENRETURN

840 L=INT(RND(1)*39)+1:IFL=20ORL=1T
HEN840

850 POKEB+L,60:GOSUB720:RETURN

860 DATA255,129,66,66,36,36,24,255

870 DATA66,126,66,66,66,126,66,66

880 DATA12,8,13,62,44,12,18,33

890 DATA24,16,24,24,24,16,16,24

900 DATA24,8,24,24,24,8,8,24

910 DATA24,8,88,62,26,24,36,66

920 DATA0,0,0,0,0,0,0,0

930 DATA129,66,66,66,98,34,34,34

940 DATA27,10,27,17,27,0,0,0

950 DATA59,10,11,9,11,0,0,0

960 DATA91,74,91,81,91,0,0,0,0

970 POKE53272,(PEEK(53272)AND240)+1
2

971 POKE56334,PEEK(56334)AND254

972 POKE1,PEEK(1)AND251

973 FORI=0TO511:POKEI+12288,PEEK(I+
53248):NEXT

974 POKE1,PEEK(1)OR4

975 POKE56334,PEEK(56334)OR1

976 RESTORE:FORI=12288+53*8TO12288+
64*8:READA:POKEI,A:NEXT

990 GOTO20

1000 FORI=1302TO1422STEP40:POKEI,54
:NEXT:FORI=1425TO1625STEP40:POKEI,5
4:NEXT

1010 FORI=1702TO1862STEP40:POKEI,54
:NEXT

1015 FORI=55574TO55694STEP40:POKEI,
3:NEXT

1020 FORI=55697TO55897STEP40:POKEI,
3:NEXT:FORI=55974TO56134STEP40:POKE
I,3:NEXT

1030 POKE1865,54:POKE1905,54:POKE56
137,3:POKE56177,3:RETURN

2000 POKES+4,0:POKES+5,0:POKES+6,0:
RETURN

2001 POKES+6,15:POKES+4,129:POKES+5
,132:POKES+6,132

2002 H3=10:L3=143:POKES+1,H3:POKES,
L3:FORT=1TO1000:NEXT:GOSUB2000:RETU
RN

3000 PRINT"[ CLR ][CRSRD][CRSRD]PAR
A PUNTUAR, DEBE SALTAR SOBRE LOS
  AGUJEROS PARA QUE EL HOMB";

3002 PRINT"RE ESTE EN      LA MEJOR
POSICION SOBRE LOS             AGUJ
EROS"

3010 PRINT"[CRSRD][CRSRD]SE INCREME
NTA EL PUNTAJE CUANDO SE       COMPL
ETA LA PANTALLA

3020 PRINT"[CRSRD][CRSRD]25 PTS POR
 AGUJERO (PANTALLA 1)"

3025 PRINT"[CRSRD][CRSRD]75 PTS POR
 AGUJERO (PANTALLA 2)"

3030 PRINT"[CRSRD][CRSRD]125 PTS PO
R AGUJERO (PANTALLA 3)"

3040 PRINT"[CRSRD][CRSRD][CRSRD]PAR
A SALTAR PULSE 'ESPACIO'"

3050 PRINT"PULSE ESPACIO PARA SEGUI
R"

3060 GR=PEEK(197):IFGR<>60THEN3060

3070 RETURN

READY.
```

rutina para el control de bloques libres en un disco

por **JESÚS DE PABLOS**

En multitud de ocasiones, nos sería muy conveniente disponer de una rutina que nos informase del número de bloques libres de que disponemos en un disco, siempre y cuando esa rutina fuera rápida y corta.

Existen métodos de exploración de catálogo que nos resuelven este problema pero su lentitud y tamaño los hacen inviables.

Hemos comprobado que el DOS de las unidades de discos COMMODORE 8050, 8250, 9060 y 9090 memoriza en unas posiciones determinadas esta información. De esta forma, a través de la función «M-R», se consigue conocer los bloques libres de un disco en un tiempo aproximado de .05 segundos según nuestras pruebas.

La aplicación más útil de esta rutina es la de incorporarla a los programas de altas de ficheros. De esta forma se puede llegar a reparticiones dinámicas del espacio en disco, sin tener que modificar los programas porque nos hemos quedado sin espacio en disco.

Aconsejamos que en la rutina de control, si se utiliza para limitar el número de fichas en un disco, no se pueda dejar el disco con menos de 50 bloques libres. Ésta no es una limitación de nuestra rutina sino un consejo para todas las unidades COMMODORE, en general, que se utilicen para la gestión de ficheros.

La única limitación de esta rutina es que deberá de mantenerse abierto durante todo el programa el canal de errores al fichero 15. ■

LISTADO

```
10   REM "PROGRAMA 'BLOQUES' : CONTROL DE BLOQUES
     LIBRES"
20   REM "AUTOR : JESÚS DE PABLOS"
30   REM "CON AUTORIZACIÓN : COPERSA - VALLADOLID"
100  CLOSE15 : OPEN15,8,15: REM "ABRIR DONDE SE QUIERA"
1000 DR=0 : GOSUB 5000 : REM "EN DR EL DRIVE DESEADO"
1010 PRINT "BLOQUES LIBRES : "+STR$(BF)
2000 CLOSE 15 : END : REM "CERRAR ANTES DE SALIR DEL
     PROGRAMA"
5000 REM "RUTINA DE BLOQUES LIBRES; EL RESULTADO
     EN BF"
5010 PRINT #15,"M-R"CHR$(157+DR)CHR$(67):GET #15,A1$
5020 PRINT #15,"M-R"CHR$(159+DR)CHR$(67):GET #15,A2$
5030 BF = ASC( A1$ + CHR$(0) ) + 256 * ASC( A2$ +
     CHR$(0) )
5040 RETURN
```

MARKETCLUB

- CBM 4.032. Intercambio programas. José Marcé Mestres. Calle Sevilla, 5. Tel. (93) 803 77 51, de 8 a 3. VILANOVA DEL CAMÍ (Barcelona).
- ACCESORIOS VIC-20: Ampliación de memoria 16K más varios programas, 13.000. Módulo de expansión para 6 cartuchos, 10.000. Cartucho lenguaje FORTH y manual, 7.000. Las tres cosas sólo por 25.000. Jaime. Tel. 245 46 56. BARCELONA.
- En Barcelona, clases de informática. PLAZAS LIMITADAS. Lenguaje BASIC. Prácticas con micro-ordenador VIC-20. Prof. E. Martínez de Carvajal. Información: Tel. (93) 345 10 00. Señorita María José (mañanas) o (93) 345 87 75. Sr. Martínez (fuera de horas de oficina).
- Desearía vender por 40.000 ptas.: VIC-20, con cartucho Super Expander, las dos partes del Curso, el «joy-stick» y un juego Indescomp, todo comprado en diciembre del 82. Fernando Martínez, calle La Roda, 39, 5º D. Teléfono 23 41 82. ALBACETE.
- Tengo un PET 2001/8K y desearía tener el cassette «Monitor para lenguaje Máquina». También desearía contactar con usuarios o clubs del PET si los hay. José Manuel Cámara Mas, calle Castor, 32, bloque II, 3º, puerta 1. ALICANTE.
- Programación de ordenadores personales; organización explotación de ficheros; programas ordenadores auxiliares, para cuestiones empresariales, profesionales, administrativas, científicas. Mora Mas. Carlos III, 41. Teléfono 339 98 29. BARCELONA-28.
- Vendo impresora Seikosha GP-100-VC para VIC-20, Commodore 64. Como nueva: 43.000 ptas. Regalo muchos programas gráficos al comprador. Llamar horas comida y cena. Fco. Gutiérrez. Tel. 253 13 40. Santiago Rusiñol, 12. MADRID-3.
- Vendo VIC-20 con expansión 16K, superexpander, curso de programación BASIC (los dos tomos), guía de referencia del programador y cartucho de juegos. Todo en perfecto estado. Precio: 55.000 ptas. Ángel Jiménez. Teléfono (954) 458 320. SEVILLA.
- Vendo VIC-20, ampl. de memoria 16K RAM, cassette. Libros: Manual VIC-20, Reference's, VIC Revealed y 25 programas muy buenos: «Comecocos», «Blitz», «S. de Ecuaciones», «Matrices», etc. Por sólo 55.000 ptas. David. Badalona, 102. Tel. (91) 734 11 03. MADRID-34.

EJEMPLARES ATRASADOS DE «CLUB COMMODORE»

Para poder satisfacer la creciente demanda de números atrasados de nuestra Revista, agotada en todas sus ediciones, hemos puesto en marcha un Servicio para suministrar fotocopias de los ejemplares que nos sean solicitados. Para recibir las fotocopias de una o de varias ediciones, no hay más que enviarnos el boletín con los datos indicados.

SERVICIO DE FOTOCOPIAS

NÚMERO DE LA EDICIÓN SOLICITADA

0	1	2	3	4	5	6	7	8	9	10	11	12

(Poner una X debajo del número de edición pedido)

Peticionario: D. ....

Calle .... n.° ....

Población .... D.P. ....

Provincia ....

Forma de envío: contra-reembolso **Precio de la edición fotocopiada: 250 ptas.**

DESCRIPCIÓN ALFABÉTICA DE LOS MNEMÓNICOS DEL 6502/6510 (II)

BCS

Salta si acarreo activado

Operación: Salta si C = 1

(Ref.: 4.1.1.4)

N Z C I D V
\- - - - - -

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
RELATIVO	BCS Oper.	B0	2	2*

* Suma 1 si se salta a la misma página.
* Suma 2 si se salta a otra página.

BEQ

Salta si el resultado es cero

Operación: Salta si Z = 1

(Ref.: 4.1.1.5)

N Z C I D V
\- - - - - -

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
RELATIVO	BEQ Oper.	F0	2	2*

* Suma 1 si se salta a la misma página.
* Suma 2 si se salta a otra página.

BIT

Compara los bits de memoria con acumulador

Operación: $A \wedge M$, $M7 \rightarrow N$, $M_6 \rightarrow V$
Los bits 6 y 7 se transfieren al registro de estado. Si el resultado de $A \wedge M$ es cero entonces Z = 1, sino Z = 0.

(Ref.: 4.2.1.1)

N Z C I D V
M7* √ - - M_6

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Pág. Cero	BIT Oper.	24	2	3
Absoluto	BIT Oper.	2C	3	4

BMI

Salta si el resultado es negativo

Operación: Salta si N = 1

(Ref.: 4.1.1.1)

N Z C I D V
\- - - - - -

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Relativo	BMI Oper.	30	2	2*

* Suma 1 si se salta a la misma página.
* Suma 2 si se salta a otra página.



Microprocesador: 6502 de MOS TECHNOLOGY de 8 bits.
Memoria: 5 Kbytes de RAM ampliables a 32 K 20 Kbytes de ROM ampliables a 28 K.
Pantalla: 23 líneas de 22 caracteres. Modulador para conectar a un televisor normal. Salida monitor video.
Colores: 8 para el marco, 16 para el fonde de la pantalla y 8 para los caracteres individuales, video inverso.
Gráficos: Semi-gráficos por teclado y alta resolución por redefinición del generador de caracteres (situándolo en RAM). Definición de 176 por 184 puntos.
Teclado: Tipo QWERTY de 62 teclas más cuatro de función definibles por el usuario.
Sonido: Tres voces de tres octavas cada una decaladas una octava entre sí, resultando una extensión total de cinco octavas. Un generador de ruido aleatorio afinable para efectos especiales, un control general de volumen.
Programación: Lenguaje BASIC, intérprete residente en ROM de 8K. Posibilidad de interceptar las funciones del Basic para crear nuevas instrucciones "a medida". El Basic del Vic es uno de los rápidos actualmente en el mercado.
Complementos: Port de usuario de 8 bits entrada/salida más dos señales de sincronismo.
Bus de expansión para ampliaciones de memoria y periféricos.
Port de juegos con conexión para dos potenciómetros (paddles), y una palanca de juegos (joystick).
Almacenamiento de masa: Unidad de cassette C2N de diseño especial para registrar programas y datos.
Ampliación de memoria: En caso de ser necesario conectar más de un cartucho al mismo tiempo, está disponible un módulo (VIC 1020) que permite la conexión simultánea de hasta seis cartuchos.

VIC-1541 UNIDAD DE DISCO

Capacidad total: 174848 bytes por disco.
Secuencial: 168656 bytes por disco.
Entradas de directorio: 144 por disco.
Sectores por pista: De 17 a 21.
Bytes por sector: 256.
Pistas: 35.
Bloques: 683 (644 bloques libres).
Soportes de información: Discos standar de 5 1/4 pulgadas, de una sola cara y densidad simple.
Sistema operativo: DOS de COMMODORE inteligente (tiene procesador propio y no ocupa memoria del ordenador central).

VIC-1525 IMPRESORA

Método de impresión: Matriz de 5 × 7 puntos, impacto por un solo martillo.
Modo caracteres: Mayúsculas y minúsculas, símbolos, números y caracteres gráficos del VIC-20.
Modo gráfico: Puntos direccionables (bit image). Siete puntos verticales por columna, 480 columna máximo.
Velocidad: 30 caracteres/segundo, de izquierda a derecha, unidireccional.
Caracteres/Línea: Máximo 80. (Posibilidad de impresión en doble ancho).
Espaciado entre líneas: 6 líneas/pulgada —modo caracteres, 9 líneas/pulgadas— modo gráfico.
Alimentación de papel: Arrastre por tractor.
Ancho de papel: Entre 4,5 y 10 pulgadas.
Copias: Original más dos copias.

CARTUCHOS

Ayuda programador: Facilita la edición y depuración de programas en Basic. Instrucciones y comandos: RENUMBER, MERGE, FIND, CHANGE, DELETE, AUTO, TRACE, STEP, OFF, KEY, EDIT, PROG, DUMP, HELP y KILL.

Super expander: Intercepta el Basic del VIC permitiendo incrementar sus instrucciones y comandos en aplicaciones gráficas de sonido y juegos. Instrucciones y comandos: KEY, GRAPHIC, COLOR, POINT, REGION, DRAW, CIRCLE, PAINT, CHAR, SCNCLR, SOUND, RGE, RCOLR, RDOT, RPOT, RPEN, RJOY y RSND.
Monitor de lenguaje máquina: Facilita enormemente la depuración de programas en lenguaje máquina, es ideal como complemento del Basic para redactar y poner en marcha rutinas de alta velocidad y manejo de datos en tiempo real. Instrucciones y comandos: ASSEMBLE, BREAKPOINT, DISASSEMBLE, ENABLE, VIRTUAL ZERO PAGE, FILL MEMORY, GO, HUNT, INTERPRET, JUMP TO SUBROUTINE, LOAD, MEMORY, NUMBER, QUICK TRACE, REGISTERS, REMOVE BREACPOINTS, SAVE, TRANSFER, WALK y EXIT TO BASIC.
Además existen cartuchos de ampliación de memoria de 3,8 y 16 Kbytes.

CURSO DE INTRODUCCION AL BASIC PARTE I y II.

En forma de libro se ha editado la primera y segunda parte de un curso de Basic que parte "de cero" y está basado en el VIC-20. Van acompañados de dos cassettes con programas y ejercicios para autocontrol.

PLOTTER VIC 1520

Método de impresión: Dibujo mediante bolígrafos de diseño especial.
Color: 4 colores; negro, azul, verde y rojo con cambio desde programa.
Cabezal: Ploter X-Y tipo tambor.
Velocidad de impresión: Media de 14 car./seg.
Caracteres por línea: Máximo 80 carac., formatos de 80, 40, 20 y 10 carac./línea.
Juego de caracteres: 96.
Velocidad de dibujo: 264 pasos/seg.
Longitud del paso: 0,2 mm. en dirección X e Y.
Velocidad de dibujo de línea: 52,8 mm./seg. en dirección X e Y. 73 mm./seg. en una línea a 45 grados.
Area de dibujo: 480 pasos (96 mm.) en dirección X. Programable en dirección Y (Máx. + — 999 de una sola vez).
Papel: Rollo de 4,5 pulgadas (114 mm.).

MONITOR EN COLOR C-1701

Pantalla: 13 pulgadas (330 mm.).
Capacidad de representación: 25 líneas de 40 caracteres.
Resolución: 320 líneas horizontales.
Compatibilidad: VIC-20 y COMMODORE 64.
Conectable a un registrador de video.
Amplificador y altavoz: Incorporados.

